Bipolarity in the Querying of Temporal Databases

Christophe Billiet(1), **José Enrique Pons**(2), **Olga Pons**(2), **Guy De Tré**(1)

(1) Dept. of Telecommunications and Information Processing
Ghent University
Sint-Pietersnieuwstraat 41, B-9000, Ghent, Belgium
Christophe.Billiet@UGent.be, Guy.DeTre@UGent.be

(2) Dept. of Computer Science and Artificial Intelligence
University of Granada
C/ Periodista Daniel Saucedo Aranda, S/N, E-18071, Granada, Spain
jpons@decsai.ugr.es, opc@decsai.ugr.es

Abstract

A database represents part of reality by containing data representing properties of real objects or concepts. To many real-world concepts or objects, time is an essential aspect and thus it should often be (implicitly) represented by databases, making these temporal databases. However, like other data, the time-related data in such databases may also contain imperfections such as uncertainties. One of the main purposes of a database is to allow the retrieval of information or knowledge deduced from its data, which is often done by querying the database. Because users may have both positive and negative preferences, they may want to query a database in a bipolar way. Moreover, their demands may have some temporal aspects. In this paper, a novel technique is presented, to query a valid-time relation containing uncertain valid-time data in a heterogeneously bipolar way, allowing every elementary query constraint a specific temporal constraint.

Keywords: Bipolar Querying, Temporal Databases, Valid-time Relation, Uncertainty, Ill-known time intervals

1 Introduction

Databases are collections of data. These data are usually the result of measurements or descriptions of aspects or properties of real-world objects or concepts. As these data represent properties of objects or concepts in reality, the database itself represents a part of reality [3], [16].

To many real-world concepts or objects, time is an essential aspect. E.g., certain historical events are meaningless without time frame. Therefore, many databases contain data representing temporal notions which describe the temporal properties of real-world objects or concepts. In the context of database systems, such temporal values [11], [4] can be classified into different categories, based on their interpretation and purpose:

- **valid-time indications** describe when a database fact or data is a true or valid representation of the reality modelled by the database [11], [4].
- **transaction-time indications** describe when a fact or data is current in the database, which means it is not logically deleted and can thus be retrieved [11], [4].
- **decision-time indications** describe when certain events were decided to happen [15].

A lot of data is made by humans. Human-made data is prone to imperfections: it can be vague or imprecise, may contain contradictions, be incomplete or contain uncertainties [16], [17]. Uncertainties in data may be caused by variability in the outcomes of an experiment and confidence in the context of such uncertainty may be modelled using probability theory [5]. Uncertainties in data may also be caused by a (partial) lack of knowledge: the exact value which is the answer to a certain question might not be known, even if there is just one correct answer and as such no variability. Confidence in the context of this kind of uncertainty may be modelled using possibility theory [19], [9], [5], [16], [17]. Uncertainties caused by a (partial) lack of knowledge may exist in temporal notions [17], [16], [5]. E.g., consider a medieval document defining a legal act. The document contains the date defining the day on which the legal act took effect. Consider the document damaged, making this date unreadable. In this case, there is uncertainty about the exact day on which the legal act took effect and this uncertainty is caused by the lack of some knowledge: it is known that there is exactly one day on which the act took effect, but it is not known which one.

One of the most important purposes of a database is of course to allow the retrieval of information and knowledge deduced from its data. Often, this is done by querying the database. Databases can be queried in a 'regular' way: the user describes the data which are desired or satisfactory to him or her and which he or she thus wants to retrieve, by perfectly describing the allowed values of this data for certain attributes in clearly stated user preferences. However, databases can also be queried in a 'fuzzy' way: the user describes the data which are desired or satisfactory to him or her by imperfectly describing the allowed values of this data for certain attributes [6]. These imperfect descriptions may contain vagueness or imprecision, often through the use of linguistic terms [13]. Databases can also be queried in what is called a 'bipolar' way. Generally, there are two main approaches to this. One is for the user to describe the data which are required (acceptable) to him or her and to describe the data among the required data, which are really desired (wished-for) to him or her, both by describing the allowed values of this data for certain attributes [10]. In the other, the user describes the data which are desired or satisfactory to him or her and the data which are undesired or unsatisfactory to him or her, both by describing the allowed values of this data for certain attributes [7], [14]. These descriptions might or might not contain imperfections. The presented work will only consider the latter approach to bipolar querying.

Generally, in querying, temporal data are handled specifically. Because of the temporal capacity and interpretation of temporal data, users usually like expressing their temporal constraints or preferences using a specific set of temporal operators [12], [17], [18]. Often, such temporal operators are based on the possible temporal relationships between two temporal values [12], [17], [18]. Such temporal relationships express semantically meaningful relationships between two temporal values. In [1] a groundbreaking collection of temporal relationships between two time intervals (and as a special case time instants or time points) [11], [4] is presented. To query temporal data in an imperfect way, fuzzy variants of temporal operators are of course necessary. Thus, several proposals have considered fuzzy variants of temporal relationships [18], [5] and of temporal operators used for the fuzzy querying of temporal databases [12], [16].

Several existing proposals have covered the regular or fuzzy querying of valid-time relational databases containing uncertain valid-time data [17], [16], [5]. However, to the knowledge of the authors, few proposals have considered the bipolar querying of valid-time databases and even fewer the bipolar querying of valid-time databases containing temporal data subject to uncertainty. The presented work tries to fill part of this gap by presenting a novel technique to query a valid-time relation containing uncertain valid-time data in a bipolar way. This novel technique allows the specification of a valid-time constraint related to each bipolar elementary query constraint. This novel technique is presented in Section 3. In Section 2, some necessary preliminary concepts and techniques are clarified and in Section 4, the conclusions of this paper and some directions for future research are given.

2 Preliminaries

In this section, some preliminary concepts and techniques are presented and briefly explained.

2.1 Valid-time relations

The presented work considers the relational database model. Here, a *relation* is composed of a *heading* and a *body*. The heading, or relation schema, is a set of attributes and describes a group of similar real-world concepts or objects. These attributes describe different aspects of these concepts or objects. For example, the heading of a relation describing wanted criminals might contain the attributes 'Date of Birth (DoB)' and 'Area of Operation (AoO)', describing the date of birth and the main city of operation of a criminal. The body is a set of n-tuples, where n is the number of attributes in the heading. Every tuple contains n values (each corresponding to an attribute) and represents one real-world object or concept in the group described by the heading. To achieve this, every such value represents the result of a measurement or description of the aspect described by its corresponding attribute, of the real-world object or concept corresponding to its tuple.

The heading of a *valid-time relation* also contains attributes describing a single valid-time indication for each tuple. These are called 'valid-time attributes'. Its body may contain several different tuples corresponding to the same real-world concept or object. Each such tuple then represents a 'state' or 'version' of the real object or concept which was or is valid (true, in existence,. . .) during the period in time given by the valid-time indication represented by the valid-time attribute values. Thus, the results of the measurements or descriptions represented by the values in such a tuple were or are true for the object or concept corresponding to the tuple, during the period in time given by the valid-time indication represented by the tuple's valid-time attribute values. In the presented work, such valid-time indications will always be time intervals [11] [4] and will always be referred to as *valid-time intervals*.

2.2 Possibilistic Variables, Ill-known Values and Ill-known Valid-time Intervals

The presented work allows the valid-time intervals of a valid-time relation's tuples to be subject to uncertainties caused by a (partial) lack of knowledge. To accomplish this, these valid-time intervals are allowed to be *ill-known intervals* [3]. In this section, possibilistic variables and the concepts of ill-known values and ill-known intervals are introduced, based on [3], [5], [8]. These concepts rely heavily on possibility theory [9]. In this work, 'possibility' is always interpreted as a measure of plausibility, given a (partial) lack of knowledge. A *possibilistic variable* is defined as follows [3], [5], [16], [9], [8].

Definition 1 *A* possibilistic variable *X on a universe U is defined as a variable taking exactly one value in U, but for which this value is (partially) unknown. The variable's possibility distribution π_X on U models the available knowledge about the value that X takes: for each $u \in U$, $\pi_X(u)$ represents the possibility that X takes the value u. This possibility is interpreted as a measure of how plausible it is that X takes the value u, given (partial) knowledge about the value X takes.*

An *ill-known value* is now defined as follows [3], [5], [16], [9], [8]:

Definition 2 *Consider a set R containing single values (and not collections of values). When a possibilistic variable X_v is defined on such a set R, the unique value X_v takes, which is (partially) unknown, is called an* ill-known value *in this work.*

An *ill-known interval* and an *ill-known valid-time interval* are now defined as follows [3], [17], [16], [8]:

Definition 3 *Consider a set R containing single values and its powerset $\wp(R)$. Now consider a subset $\wp_I(R)$ of $\wp(R)$ and let this subset contain* *every element of $\wp(R)$ that is an interval, but no other elements. When a possibilistic variable X_i is defined on the subset $\wp_I(R)$ of the powerset $\wp(R)$ of some set R, the unique value X_i takes will be a crisp interval and the possibility distribution π_{X_i} of X_i will be a possibility distribution on $\wp_I(R)$. This π_{X_i} will define the possibility of each value of $\wp_I(R)$ (a value of $\wp_I(R)$ is a crisp interval in R) being the value X_i takes. This exact value the variable takes, is called an* ill-known interval *here. Seen as the ill-known interval basically defines an interval in R, it is also called an ill-known interval in R in this context. When the set R is an ordered set of time points which is represented by the domain of a set of valid-time attributes, such ill-known interval will be called an* ill-known valid-time interval *in the presented work.*

The definition above uses the most general approach to defining ill-known valid-time intervals. However, in the presented work, another approach will be used. Here, an ill-known valid-time interval is defined by its start and end point, which are ill-known values in the time domain [11], [4] represented by the valid-time attributes' domain. An ill-known valid-time interval is seen as an interval of which the exact start and end point are (partially) unknown, which implies that the interval itself is (partially) unknown.

Investigating the differences and similarities between these approaches and the correspondences, interactions and transformations between them are part of the current research of the authors.

In the presented work, the possibility distributions defining the ill-known values which define an ill-known valid-time interval, will take a triangular shape [17], [16]. This means, for a possibility distribution π_X on an ordered set of time points T, there will be a single value $m \in T$ for which $\pi_X(m) = 1$, there will be a single value $m - a \in T$, determined by the distance a, for which $m - a = \inf\{u \in T : \pi_X(u) > 0\}$, there will be a single value $m + b \in T$, determined by the distance b, for which $m + b = \sup\{u \in T : \pi_X(u) > 0\}$ and every value in T will comply with:

$$\pi_X(u) = \begin{cases} 0 & if\ u \leq m - a \\ \dfrac{u - m + a}{a} & if\ u \geq m - a\ and\ u \leq m \\ 1 & if\ u = m \\ \dfrac{m - u}{b - m} + 1 & if\ u \geq m\ and\ u \leq m + b \\ 0 & if\ u \geq m + b. \end{cases}$$

ID	IID	DoB	AoO	VST	VET
1	1	11/08/77	Bruges	$[5/11/02, 1, 2]$	$[10/11/02, 1, 1]$
1	2	11/08/77	Ghent	$[16/11/02, 1, 1]$	$[24/11/02, 4, 3]$
1	3	11/08/77	Antwerp	$[10/06/03, 5, 3]$	$[11/08/04, 2, 2]$
2	1	23/05/77	Antwerp	$[16/11/02, 1, 1]$	$[24/11/02, 4, 2]$
2	2	23/05/77	Ghent	$[20/06/03, 4, 7]$	$[11/08/04, 4, 7]$

Table 1: A visualisation of an example relation, containing information on criminals. The value for attribute ID uniquely identifies a criminal, the combination of values for attributes ID and IID uniquely identify a state or version of a criminal.

The triangular shape described above will allow such possibility distribution and thus its corresponding ill-known value to be totally defined and described by its values m, a and b alone. Therefore, such possibility distribution will be noted $[m, a, b]$ in this paper. Crisp time points (and thus crisp time intervals) can now still be defined by setting their values a and b to 0.

Thus, in what follows, the valid-time indications in the valid-time relation will be ill-known valid-time intervals. For reasons of convenience, these intervals will be described by two valid-time attributes: a so-called 'valid start time' (VST), which describes the ill-known value which is the interval's start point and a so-called 'valid end time' (VET), which describes the ill-known value which is the interval's end point. All of this is illustrated in Table 1, which is a visualisation of the example relation which will be used throughout the rest of the paper.

2.3 Ill-known Constraints

In the presented paper, the specific evaluation of a temporal constraint for a given record is done using the framework of *ill-known constraints* [5], [16], [17]. In [5], the notion of an ill-known constraint is introduced:

Definition 4 *Given a universe U, an ill-known constraint C is specified by means of a binary relation $R \subseteq U^2$ and a fixed, ill-known value defined by its possibilistic variable V on U, i.e.,*

$$C \triangleq (V, R).$$

A set $A \subseteq U$ now satisfies this constraint C if and only if:

$$\forall a \in A : (V, a) \in R.$$

An example of an ill-known constraint is $C_> \triangleq (X, >)$. A set A then satisfies $C_>$ if $\forall a \in A : X > a$.

Basically, the satisfaction of a crisp constraint by a set A is a boolean matter (A either satisfies the constraint or not) and can be seen as a boolean proposition. However, for an ill-known constraint $C \triangleq (V, R)$, due to the uncertainty inherent to the ill-known value V, it can be uncertain whether C is satisfied by A or not [5], [3]. Based on the possibility distribution π_V of V, the possibility and necessity that A satisfies C can be determined. This proposition can thus be seen as a possibilistic variable on $\mathbb{B}$. The possibility and necessity are obtained by:

$$Pos(A \text{ satisfies } C) = \min_{a \in A} \left(\sup_{(w,a) \in R} \pi_V(w) \right) \tag{1}$$

$$Nec(A \text{ satisfies } C) = \min_{a \in A} \left(\inf_{(w,a) \notin R} (1 - \pi_V(w)) \right). \tag{2}$$

Such possibility and necessity pairs can be aggregated in order to reflect the evaluation of logical compositions of ill-known constraints [5].

2.4 Bipolar Querying

As mentioned before, sometimes people express their preferences using both positive and negative statements. In some cases, the semantics of these statements are non-symmetric in such a way that the positive preferences can not be derived from the negative or vice versa. In these cases, the bipolarity in the query specification is called 'heterogeneous'. The presented work will consider such heterogeneous bipolar querying.

In database querying, bipolarity can either be specified inside elementary query conditions, or it can be specified between elementary query conditions. In [14], it is shown that combining both approaches makes no sense and, more importantly, the approach where bipolarity is specified inside elementary query conditions, using intuitionistic fuzzy sets [2], is a more intuitive one. Thus, in the presented work, we will only use the latter approach.

In this approach, elementary query conditions are allowed to express both what is accepted and what is not accepted as a result of the query, at once. Such query conditions are called 'bipolar query conditions' [14].

As described in [14], [7], a bipolar query condition c_A expressing the user's preferences about the values of an attribute A can be modelled by an Intuitionistic Fuzzy Set (IFS) [2] as:

$$c_A = \{(x, \mu_{c_A}(x), \nu_{c_A}(x)) : x \in dom_A\}.$$

Here, dom_A is the domain of attribute A's data type, the membership degree $\mu_{c_A}(x)$ corresponding to a value $x \in dom_A$ represents to what extent value x is satisfactory to the user, whereas the non-membership degree $\nu_{c_A}(x)$ corresponding to a value $x \in dom_A$ represents to what extent value x is unsatisfactory to the user [14], [7]. Note that to allow the user's preferences to be overspecified, the IFS's consistency condition can be relaxed, which means that

$$\forall x \in dom_A : \mu_{c_A}(x) + \nu_{c_A}(x) \leq 1$$

does not necessarily have to hold.

If the user defines $\mu_{c_A}(x)$, explicitly providing his or her positive preferences, but doesn't define $\nu_{c_A}(x)$, then the non-membership function will be assumed to be the inverse of the membership function, i.e.,

$$\nu_{c_A}(x) = 1 - \mu_{c_A}(x), \forall x \in dom_A.$$

If the user defines $\nu_{c_A}(x)$, explicitly providing his or her negative preferences, but doesn't define $\mu_{c_A}(x)$, then the membership function will be assumed to be the inverse of the non-membership function, i.e.,

$$\mu_{c_A}(x) = 1 - \nu_{c_A}(x), \forall x \in dom_A.$$

Thus, in the absence of clear heterogeneousness of the bipolarity, the bipolarity will be assumed homogeneous.

In the approach presented in [14], the evaluation of a bipolar query condition c_A results in a so-called *bipolar satisfaction degree* (BSD), which is a pair

$$(s, d),\ s, d \in [0, 1]$$

where s is called the *satisfaction degree* and d is called the *dissatisfaction degree*. Both s and d take their values in the unit interval $[0, 1]$ and are independent of each other: they independently denote to which extent the BSD respectively represents 'satisfied' and 'dissatisfied'. Extreme values for s and d are 0 ('not at all') and 1 ('fully'). As such and as special cases, the BSD $(1, 0)$ represents 'fully satisfied, not dissatisfied at all', whereas $(0, 1)$ represents 'not satisfied at all, fully dissatisfied'.

From a semantical point of view, BSD's are closely related to Atanassov intuitionistic fuzzy sets (AFS) [2], except that it is explicitly assumed that there is no consistency condition for BSD's, i.e., a condition like $0 \leq s+d \leq 1$ is missing. Indeed, because s and d are considered to be completely independent of each other, it is allowed that $s + d > 1$. The motivation for this is that BSD's try to reflect heterogeneous bipolarity in human reasoning, and that human reasoning can sometimes be inconsistent.

In general, the evaluation of a database tuple R against a bipolar query condition c_A over attribute A will result in a BSD, which is calculated as follows ($R[A]$ is the value of tuple R for attribute A):

$$(s^R_{c_A}, d^R_{c_A}) = (\mu_{c_A}(R[A]), \nu_{c_A}(R[A])) \tag{3}$$

with $s^R_{c_A}$ and $d^R_{c_A}$ the satisfaction degree, respectively dissatisfaction degree, of tuple R for condition c_A.

If only positive information is given by the user (μ_{c_A}), this is reduced to

$$(s^R_{c_A}, d^R_{c_A}) = (\mu_{c_A}(R[A]), 1 - \mu_{c_A}(R[A]))$$

Analogously, when only negative information is given (ν_{c_A}), this is reduced to

$$(s^R_{c_A}, d^R_{c_A}) = (1 - \nu_{c_A}(R[A]), \nu_{c_A}(R[A]))$$

Remark again that the traditional approach with regular satisfaction degrees can be obtained as a special case, namely in the case of symmetric bipolarity. In that case, one merely has to omit the dissatisfaction degree from the BSD to receive the traditional satisfaction degrees.

3 Bipolar Querying of a Valid-time Relation

In this section, a novel technique is proposed to query a valid-time relation with valid-time indications subject to uncertainty in a heterogeneously bipolar way. The novelty of this technique is that it allows a local specification of the user's temporal preferences, i.e., for every elementary query condition, the user can specify during which crisp valid-time interval this condition should hold. First, the assumed structure of the valid-time relation is discussed, next the proposals for the construction of a query and the evaluation of such a query are presented and last, some issues concerning record ranking and the technique adopted here are given.

3.1 Structure of the Relation

The presented work will consider valid-time relations with valid-time indications which are ill-known valid-time intervals, as described in Section 2.2. The example given in that section will continue to serve as an example throughout the rest of the paper.

3.2 Construction of the Query

In the current proposal, a query Q is constructed as:

$$Q = (c_{A_1}, tc_{A_1})\ op_1 \dots\ op_{n-1}\ (c_{A_n}, tc_{A_n}).$$

Here, for every $i \in \mathbb{N}$, for which $1 \leq i \leq n$, every c_{A_i} is a bipolar elementary query condition specified using an IFS (as shown in Section 2.4) and every op_i is an operator, which can be either 'AND' or 'OR'. Every tc_{A_i} is now a temporal constraint. Such a temporal constraint tc_{A_i} is now any construction which can be evaluated to a single Allen relationship and a single crisp time interval.

The interpretation here is that the user requires records for which the corresponding elementary query condition is valid during a time interval related to the time interval to which the temporal constraint evaluates. The nature of this relationship is given by the Allen relationship. Remembering that a record in a valid-time relation represents a state or version of a real object or concept (as opposed to a real object itself), the user's query demands are interpreted as demands towards the state of an object or concept during the given corresponding time period(s). This means that the user may express demands about the current state of an object or concept (by specifying a time interval containing the present) and demands about previous states of an object or concept (by specifying time intervals in history). Thus, the user can describe the current and previous states of the object or concept he or she requires and thus some kind of required 'history'. Obviously, several elementary query conditions may concern the same attribute, but indicate a different time period.

Considering the example relation given in Section 2.2, a user could be interested in identifying a criminal who 'was born somewhere in the summer of 1977, operated in the vicinity of Bruges from 6/11/02 until 10/11/02 and operated in the surroundings of Ghent, but certainly not around Bruges any more, since 16/11/02'. The corresponding query would then be:

$$Q_{ex} = (c_{DoB},\ tc_{DoB})\ AND\ (c_{AoO,1},\ tc_{AoO,1})\ AND\ (c_{AoO,2},\ tc_{AoO,2})$$

where

- $(c_{DoB},\ tc_{DoB})$ models the criterion 'was born somewhere in the summer of 1977'.
  - Under the consideration that T is a time domain containing all dates in time,

    $$c_{DoB} = \{(x, \mu_{c_{DoB}}(x), 1 - \mu_{c_{DoB}}(x)) : x \in T\}$$

    with membership function

    $$\mu_{c_{DoB}}(x) = \begin{cases} 0 & if\ x \text{ is a date with month in} \\ & \quad \{Jan, Feb, Mar, Apr, Oct, Nov, Dec\} \\ 0.5 & if\ x \text{ is a date with month in } \{May, Sep\} \\ 1 & if\ x \text{ is a date with month in} \\ & \quad \{Jun, Jul, Aug\} \end{cases}$$

    and the non-membership function $\nu_{c_{DoB}}$ is the inverse of the membership function $\mu_{c_{DoB}}$.
  - $tc_{DoB} = during\,]-\infty, \infty[$, which reflects that there are no specific constraints on the valid time for this criterion.

- $(c_{AoO,1},\ tc_{AoO,1})$ models the criterion 'operated in the vicinity of Bruges from 6/11/02 until 10/11/02'. Hereby
  - $c_{AoO,1} = \{(x, \mu_{c_{AoO,1}}(x), 1 - \mu_{c_{AoO,1}}(x)) : x \in Cities\}$ where the membership function $\mu_{c_{AoO,1}}$ is the one of the fuzzy set

    $$\{(Bruges, 1), (Ghent, 0.7), (Antwerp, 0.3)\}$$

    and the non-membership function $\nu_{c_{AoO,1}}$ is the inverse of the membership function $\mu_{c_{AoO,1}}$.
  - $tc_{AoO,1} = during\,[6/11/02, 10/11/02]$ models 'from 6/11/02 until 10/11/02'.

- $(c_{AoO,2},\ tc_{AoO,2})$ models the criterion 'certainly not around Bruges any more, since 16/11/02'. Hereby
  - $c_{AoO,2} = \{(x, \mu_{c_{AoO,2}}(x), 1 - \mu_{c_{AoO,2}}(x)) : x \in Cities\}$ where the membership function $\mu_{c_{AoO,2}}$ is the one of the fuzzy set

    $$\{(Bruges, 0.3), (Ghent, 1), (Antwerp, 0.7)\}$$

and the non-membership function $\nu_{c_{AoO,2}}$ is the membership function of the fuzzy set

$$\{(Bruges, 1), (Ghent, 0.3), (Antwerp, 0.3)\}.$$

– Under the consideration that 'NOW' indicates the current date, $tc_{AoO,2} = during\,[16/11/02, NOW]$ models 'since 16/11/02'.

3.3 Query Evaluation

The query is evaluated for every record in the relation. For every record in the relation, the following happens distinctly:

- Every non-temporal elementary query condition is evaluated, resulting in a BSD for each. This evaluation is done as described in [14], using Eq. (3). The resulting BSD expresses the extend to which the record's value for the corresponding attribute satisfies and dissatisfies the user's non-temporal demand expressed in the elementary query condition.

- Every temporal constraint corresponding to an elementary query condition is evaluated, resulting in a possibility degree and a necessity degree. For this evaluation, the possibility and necessity degrees expressing respectively the possibility and necessity that the record's valid-time interval is in the given Allen relationship with the given crisp time interval are calculated using ill-known constraints. This calculation is done exactly as described in [16], [5], using Eq. (1)-(2).

The interpretation here is that the resulting BSD expresses to which extend the state of an object or concept satisfies the user's request, while the possibility and necessity degrees express how plausible, respectively necessary it is that the object or concept under consideration is in this state during a time period related to the time period specified by the user in the manner specified by the user.

In Table 2, the resulting BSD's, possibilities and necessities after evaluation of the individual criteria in the example query for the records of Table 1 are shown.

3.4 Object Ranking

The purpose of evaluating a query is of course to provide the user with the objects or concepts most fitting to his or her needs. In this case, two different criteria play a role.

ID	IID	c_{DoB}	tc_{DoB}	$c_{AoO,1}$	$tc_{AoO,1}$	$c_{AoO,2}$	$tc_{AoO,2}$
1	1	(1,0)	(1,1)	(1,0)	(1, 0)	(0.3,1)	(0,0)
1	2	(1,0)	(1,1)	(0.7, 0.3)	(0, 0)	(1,0.7)	(1,0)
1	3	(1,0)	(1,1)	(0.3, 0.7)	(0, 0)	(0.7,0.3)	(1,1)
2	1	(0.5, 0.5)	(1,1)	(0.3, 0.7)	(0, 0)	(0.7,0.3)	(1,0)
2	2	(0.5, 0.5)	(1,1)	(0.7, 0.3)	(0, 0)	(1,0.7)	(1,1)

Table 2: The example query evaluation for the example table. The values for the non-temporal constraints show BSD's (s, d), the others pairs (p, n) consist of a possibility degree p and necessity degree n.

1. The possibility and necessity degrees constitute quantifications of confidence in a context of valid-time uncertainty and thus portray the confidence in and necessity of the presence of an object able to fulfill the user's requests. These quantifications answer the question: 'How plausible is it that a suitable object or concept is available?'.

2. The (dis)satisfaction degrees constitute quantifications of satisfaction and dissatisfaction and thus portray the level of (dis)satisfaction an object could bring the user with respect to his or her demands. These quantifications answer the question: 'To what extend would a possibly available object (dis)satisfy the user's demands?'.

A fundamental question poses itself now: how can both quantifications be combined so as to obtain a single ranking of the results? An unambiguous and straightforward ranking allows to easily present the query results best fitting the user's demands. When ranking the results, the importance the user allocates to availability and (dis)satisfaction should be carefully examined and taken into account: some users might not care so much about availability, as long as they are sufficiently satisfied with the object, or vice versa. It is important to keep both quantifications as separate (meta)data in the ranked results presented to the users, else the mutually different interpretations of both quantifications would be lost.

In most existing proposals dealing with a similar situation, both quantifications are combined as to restrict each other. The result is generally seen as a quantification of the possibility that the user requirements are met. In the presented work, this same approach will be followed.

For every couple (c_{A_i}, tc_{A_i}) of a non-temporal elementary query constraint c_{A_i} and the corresponding temporal constraint tc_{A_i}, let

- $(s^R_{c_{A_i}}, d^R_{c_{A_i}})$ be the BSD and

- $(pos^R_{tc_{A_i}}, nec^R_{tc_{A_i}})$ be the possibility and necessity pair

all resulting from the evaluation of (c_{A_i}, tc_{A_i}) for a database tuple R. First, a score

$$sc^R_{c_{A_i}} = \frac{(s^R_{c_{A_i}} - d^R_{c_{A_i}} + 1)}{2}$$

is calculated, expressing how well the record fulfils the positive and negative non-temporal user demands about attribute A_i. This calculation is based on a scoring function suggested in [14] (but rescaled to cover the unit interval) and could be replaced by another consistent one. Now, the possibility $pos^R_{A_i}$ and the necessity $nec^R_{A_i}$ that the user's requirements about A_i are met, are calculated as follows:

$$\begin{aligned} pos^R_{A_i} &= \min(sc^R_{c_{A_i}}, pos^R_{tc_{A_i}}) \\ nec^R_{A_i} &= \begin{cases} 0 & \text{if } pos^R_{A_i} < 1 \\ \min(sc^R_{c_{A_i}}, nec^R_{tc_{A_i}}) & \text{else.} \end{cases} \end{aligned}$$

Every database tuple R represents an object or concept state. Let

$$\{R_{o,i} : i \in \mathbb{N} \wedge 1 \leq i \leq m\}$$

be the set of tuples $R_{o,i}$ representing states of object o. Then, for every such couple (c_A, tc_A), for every object or concept o, the degrees $pos_A^{R_{o,i}}$ and $nec_A^{R_{o,i}}$, $i \in \mathbb{N} \wedge 1 \leq i \leq m$ must be combined in a general possibility degree pos^o_A, respectively necessity degree nec^o_A, to express how possible (resp. necessary) it is that o meets the user's demands about A. For this, a maximum function is used, to express that if any state of o has a high plausibility of meeting the user's demands, then o should be seen similar. Thus:

$$\begin{aligned} pos^o_A &= \max_{1 \leq i \leq m} (pos_A^{R_{o,i}}) \\ nec^o_A &= \max_{1 \leq i \leq m} (nec_A^{R_{o,i}}). \end{aligned}$$

The results of these calculations for the example are shown in Table 3. Based on these possibility and necessity degrees, a consistent ranking can be made easily. In the context of the presented work, it is suggested to model the 'AND' query operator with a minimum function and the 'OR' query operator with a maximum function. This would result in the objects with ID's respectively 1 and 2 having final possibilities 0.7, resp. 0 and final necessities 0, resp. 0.

ID	pos_{DoB}	nec_{DoB}	$pos_{AoO,1}$	$nec_{AoO,1}$	$pos_{AoO,2}$	$nec_{AoO,2}$
1	1	1	1	0	0.7	0
2	0	0	0	0	0.7	0

Table 3: The resulting possibility and necessity degrees.

4 Conclusions

In this paper, a novel technique is presented, to query a valid-time relation containing uncertain valid-time data in a heterogeneously bipolar way, allowing every elementary query constraint a specific temporal constraint. Furthermore, a major issue in combining quantifications of (dis)satisfaction with quantifications of confidence in a context of partial knowledge is described and shortly discussed. In the near future the possible interactions between valid-time uncertainty and bipolar querying will be further explored and some considerable attention will be dedicated to the issue in combining semantically different quantifications.

References

[1] Allen, J. F. Maintaining Knowledge about Temporal Intervals. *Communications of the ACM 26*, 11 (1983), 832–843.

[2] Atanassov, K. T. Intuitionistic Fuzzy Sets. *Fuzzy Sets and Systems 20*, 1 (1986), 87–96.

[3] Billiet, C., Pons, J. E., Pons Capote, O., and De Tré, G. Evaluating Possibilistic Valid-Time Queries. In *CCIS* (2012), vol. 297, pp. 410–419.

[4] Böhlen, M. H. e. a. The Consensus Glossary of Temporal Database Concepts-February 1998 Version. In *LNCS* (1998), vol. 1399/1998, pp. 367–405.

[5] Bronselaer, A., Pons, J. E., De Tré, G., and Pons, O. Possibilistic evaluation of sets. *Accepted for publication in the International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* (2013).

[6] De Caluwe, R., and De Tré, G. Preface to the Special Issue on Advances in Fuzzy Database Technology. *International Journal of Intelligent Systems 22*, 7 (2007), 661–663.

[7] De Tré, G., Zadrozny, S., and Bronselaer, A. J. Handling Bipolarity in Elementary Queries to Possibilistic Databases. *IEEE Transactions on Fuzzy Systems 18*, 3 (2010), 599–612.

[8] Dubois, D., and Prade, H. Incomplete Conjunctive Information. *Computers & Mathematics with Applications 15*, 10 (1988), 797–810.

[9] Dubois, D., and Prade, H. *Possibility Theory*. Plenum Press, 1988.

[10] Dubois, D., and Prade, H. Bipolarity in Flexible Querying. In *LNCS* (2002), vol. 2522, pp. 174–182.

[11] Dyreson, C. E. e. a. A Consensus Glossary of Temporal Database Concepts. *SIGMOD Record 23*, 1 (1994), 52–64.

[12] Galindo, J., and Medina, J. M. FTSQL2: Fuzzy Time in Relational Databases*. In *Proceedings of the 2nd International Conference in Fuzzy Logic and Technology* (2001), pp. 47–50.

[13] Kacprzyk, J., and Zadrozny, S. Computing with Words in Intelligent Database Querying: Standalone and Internet-based Applications. *Information Sciences 134*, 1-4 (2001), 71–109.

[14] Matthé, T., De Tré, G., Zadrozny, S., and Kacprzyk, J. Bipolar Database Querying Using Bipolar Satisfaction Degrees. *International Journal of Intelligent Systems 26*, 10 (2011), 890–910.

[15] Nascimento, M. A., and Eich, M. H. Decision Time in Temporal Databases. In *Proceedings of the Second International Workshop on Temporal Representation and Reasoning* (1995), pp. 157–162.

[16] Pons, J. E., Billiet, C., Pons Capote, O., and De Tré, G. A Possibilistic Valid-Time Model. In *CCIS* (2012), vol. 297, pp. 420–429.

[17] Pons, J. E. e. a. A Relational Model for the Possibilistic Valid-time Approach. *International Journal of Computational Intelligence Systems 5*, 6 (2012), 1068–1088.

[18] Schockaert, S., De Cock, M., and Kerre, E. E. Fuzzifying Allens Temporal Interval Relations. *IEEE Transactions on Fuzzy Systems 16*, 2 (2008), 517–533.

[19] Zadeh, L. A. Fuzzy Sets as a Basis for a Theory of Possibility. *Fuzzy Sets and Systems 1*, 1 (1978), 3–28.